****

**Praca projektowa programowanie obiektowe**

*Elektroniczny system oceniania*

Prowadzący: Autor:

mgr inż. Ewa Żesławska *Krzysztof Motas*

nr albumu: 125145

Kierunek: Informatyka, grupa lab 2

Rzeszów 2024

Spis treści

[**1.** **Opis założeń projektu** 3](#_Toc155609263)

[**2.** **Specyfikacja wymagań** 5](#_Toc155609264)

[**3.** **Opis struktury projektu** 7](#_Toc155609265)

[**4.** **Harmonogram realizacji projektu** 11](#_Toc155609266)

[**5.** **Prezentacja warstwy użytkowej projektu** 13](#_Toc155609267)

[**6.** **Podsumowanie** 19](#_Toc155609268)

[**7.** **Literatura** 20](#_Toc155609269)

1. **Opis założeń projektu**

**Wstęp**

W erze dynamicznego rozwoju technologii, tradycyjne metody prowadzenia dzienników ocen w szkołach stają się coraz bardziej przestarzałe. Papierowe dzienniki, mimo że były używane przez wiele lat, nie spełniają już oczekiwań nowoczesnego systemu edukacji. Nauczyciele, którzy wciąż korzystają z tradycyjnych dzienników, często spotykają się z wieloma przeszkodami związanymi z ręcznym zarządzaniem ocenami. Przeglądanie i analizowanie ocen są czasochłonne, co wpływa na zmniejszenie efektywności procesu oceniania. Innym aspektem, który wymaga uwagi jest fakt, że uczniowie nie mają bezpośredniego dostępu do swoich postępów w nauce. Jest to ważna kwestia, która w kontekście tradycyjnych dzienników wprowadza ograniczenia w samodzielnym monitorowaniu osiągnięć edukacyjnych. W odpowiedzi na te problemy, pojawia się rozwiązanie – projekt stworzenia elektronicznego systemu oceniania.

**Cele projektu**

Nadrzędnym celem projektu jest stworzenie nowoczesnego, elektronicznego systemu oceniania, który zastąpi tradycyjne dzienniki i usprawni procesy związane z efektywnym gromadzeniem, analizą oraz prezentacją ocen uczniów. System pozwoli także zredukować koszty związane z papierowymi dziennikami, poprzez zmniejszenie zapotrzebowania na papier, drukarki oraz artykuły biurowe. Ograniczenie zużycia papieru będzie miało również pozytywny wpływ na środowisko naturalne - zmniejszenie zapotrzebowania na papier przyczynia się do ochrony drzew i redukcji emisji związków chemicznych.

**Założenia projektu**

Niezbędną czynnością do stworzenia elektronicznego systemu oceniania jest określenie funkcjonalności, które umożliwią efektywne zarządzanie ocenami cząstkowymi z różnych przedmiotów. Kluczowym elementem jest zapewnienie nauczycielom prostego narzędzia do wprowadzania oraz monitorowania ocen. Nauczyciele powinni mieć możliwość tworzenia raportów ocen dla poszczególnych uczniów, zawierających oceny cząstkowe ze wszystkich przedmiotów oraz ich średnie. Dodatkowo, istotną funkcjonalnością systemu powinna być możliwość generowania statystyk dla każdego przedmiotów. Te statystyki powinny obejmować informacje o uczniach: który uczeń posiada najniższą lub najwyższą średnią ocen z danego przedmiotu itp. To narzędzie pozwoli nauczycielom na szybką identyfikację uczniów wymagających dodatkowej uwagi lub tych, którzy osiągają wybitne wyniki. Dla uczniów kluczową funkcją systemu powinna być możliwość dostępu do listy ocen z poszczególnych przedmiotów. Lista ta powinna zawierać informacje o dacie otrzymania oceny, przedmiocie, za co została wystawiona oraz przez którego nauczyciela. Dzięki temu uczniowie będą posiadać szybki dostęp do swoich postępów w nauce.

Efektem prac będzie stworzenie aplikacji desktopowej, która kompleksowo realizuje wszystkie założenia projektu. Jej wprowadzenie przyniesie efektywność, precyzję oraz łatwość dostępu do danych.

1. **Specyfikacja wymagań**

Niniejsza specyfikacja zawiera szczegółowe wymagania funkcjonalne i niefunkcjonalne, które stanowią fundament dla projektu.

**Wymagania funkcjonalne**

Poniżej znajduje się lista kluczowych funkcji, które system powinien realizować. Opisują one operacje, usługi oraz interakcje, które będą dostępne dla użytkowników systemu.

* Logowanie użytkowników:
* Zapewnienie bezpiecznego uwierzytelniania i autoryzacji użytkowników poprzez unikalne identyfikatory (adresy e-mail) i hasła.
* Zarządzanie ocenami:
* Umożliwienie nauczycielom wprowadzanie ocen cząstkowych dla poszczególnych uczniów z różnych przedmiotów.
* Zapewnienie możliwości edycji oraz usuwania wprowadzonych ocen.
* Generowanie raportów:
* Dla nauczycieli umożliwienie generowania raportów ocen dla poszczególnych uczniów, zawierających oceny cząstkowe ze wszystkich przedmiotów oraz ich średnie.
* Dla uczniów umożliwienie dostępu do raportów z własnymi ocenami.
* Generowanie statystyk:
* Umożliwienie generowania statystyk dla każdego przedmiotu, takich jak informacje o uczniach posiadających najniższą lub najwyższą średnią ocen z danego przedmiotu.
* Zapewnienie narzędzia umożliwiającego nauczycielom szybką identyfikację uczniów wymagających dodatkowej uwagi lub osiągających wybitne wyniki.
* Dostęp do listy ocen przez uczniów:
* Zapewnienie uczniom dostępu do listy ocen z poszczególnych przedmiotów.
* Lista powinna zawierać informacje o dacie otrzymania oceny, przedmiocie, za co została wystawiona oraz przez którego nauczyciela.
* Aplikacja desktopowa:
* Stworzenie kompleksowej aplikacji desktopowej, spełniającej wszystkie założenia projektu.
* Zapewnienie efektywności, precyzji oraz łatwości dostępu do danych.

**Wymagania niefunkcjonalne**

Poniżej znajdują się założenia dotyczące jakości i wydajności systemu. Obejmują one aspekty, takie jak bezpieczeństwo, wydajność, dostępność oraz inne właściwości, które wpływają na doświadczenie użytkownika.

* Użyteczność:
* Aplikacja powinna być intuicyjna i łatwa w obsłudze zarówno dla nauczycieli, jak i uczniów.
* Estetyczna i czytelna prezencja informacji.
* Bezpieczeństwo:
* Zapewnienie bezpiecznego przechowywania danych użytkowników, w tym ocen.
* Zastosowanie mechanizmów walidacji danych.
* Wydajność:
* Aplikacja powinna działać płynnie, bez zbędnych opóźnień.
* Akceptowalne czasy odpowiedzi na żądania użytkowników.
* Optymalizacja zapytań do bazy danych dla zapewnienia płynności działania.
* Dostępność:
* System powinien być dostępny dla użytkowników w dowolnym miejscu i czasie.
* Aplikacja powinna być dostępna bez błędów na wspieranych systemach operacyjnych.
* Skawalność:
* Łatwość w dodawaniu nowych funkcji i obsługi powiększającej się liczby użytkowników bez utraty wydajności.
* Środowisko:
* Wykorzystanie najnowszych frameworków i bibliotek programistycznych do stworzenia nowoczesnej i skalowanej aplikacji desktopowej.
* Regularne aktualizacje systemu zgodnie z najnowszymi wersjami bibliotek i środowisk programistycznych.
* Utrzymanie:
* Zastosowanie nowoczesnych narzędzi do zarządzania kodem źródłowym, takich jak systemy kontroli wersji (np. Git), dla efektywnego śledzenia zmian.

1. **Opis struktury projektu**

Poniżej przedstawiono informacje dotyczące struktury projektu. Obejmują one używane środowisko programistyczne, narzędzia, minimalne wymagania sprzętowe, hierarchię klas oraz strukturę przechowywanych danych.

* Środowisko programistyczne:
* Oprogramowanie: IntelliJ IDEA 2023.2.2.
* Implementacja języka programowania: OpenJDK 21.0.2.
* Narzędzia:
* Biblioteka graficzna: Swing.
* Biblioteka Look and Feel: FlatLaf 3.3 (oraz JSVG).
* Sterownik JDBC: MySQL Connector/J 8.2.0.
* Rekomendowane minimalne wymagania sprzętowe:
* Procesor: Intel Core i3 lub równoważny.
* Pamięć RAM: Minimum 4 GB.
* Dysk twardy: Minimum 5 GB wolnego miejsca.
* System operacyjny: Windows 11.
* Struktura bazy danych:

Na rysunku nr 1 przedstawiono diagram ERD bazy danych wymaganej do poprawnego działania projektu.

Obraz zawierający tekst, zrzut ekranu, Czcionka, linia

Opis wygenerowany automatycznie

Rysunek 1. Diagram ERD bazy danych

Baza danych składa się z trzech tabel:

1. Tabela „accounts” zawiera informacje o użytkownikach, takie jak identyfikator, adres-email, hasło, imię, nazwisko i rola (student lub nauczyciel).
2. Tabela „grades” przechowuje informacje o ocenach, takie jak identyfikator oceny, identyfikator ucznia. identyfikator nauczyciela, ocenę, rodzaj oceny, data wystawienia oraz identyfikator przedmiotu.
3. Tabela „subjects” zawiera informacje o przedmiotach, takie jak identyfikator i nazwa przedmiotu.

Do poprawnego funkcjonowania projektu, konieczne jest posiadanie bazy danych. Plik .sql zawierający strukturę bazy danych znajduje się w głównym katalogu repozytorium projektu pod nazwą eso.sql. Przed rozpoczęciem korzystania z aplikacji, wymagane jest zaimportowanie tego pliku do SZBD.

* Hierarchia klas:

Na rysunku nr 2 przedstawiono diagram klas projektowanej aplikacji.

Obraz zawierający tekst, zrzut ekranu, numer, Czcionka

Opis wygenerowany automatycznie

Rysunek 2. Diagram klas aplikacji

* Klasa „Main” pełni rolę głównej klasy programu, umożliwiając uruchamianie aplikacji.
* W klasie „Database” znajduje się metoda statyczna „getConnection”, której celem jest ułatwienie nawiązywania połączenia z bazą danych, eliminując konieczność podawania danych takich jak URL, nazwa użytkownika i hasło przy każdym zapytaniu.
* Klasa „User” reprezentują użytkownika, posiadając takie pola jak „id”, „emailAddress”, „name”, „surname” oraz „role”. Dla każdego z tych pól zastosowano hermetyzację, a każde z pól posiada odpowiednie gettery i setery.
* Klasa „Role” reprezentuje rolę użytkownika, posiada metody takie jak „getId” oraz „getName”.
* Klasa „Utilities” zawiera metody pomocnicze takie jak np. „emailPatternMatches” oraz „passwordPatternMatches” umożliwiające walidację danych logowania.
* Klasa „LoginForm” umożliwia logowanie do Elektronicznego Systemu Oceniania. Użytkownikowi prezentowane jest okno logowania, a jeżeli wprowadzi on poprawne dane, zostanie zalogowany, co skutkuje wywołaniem obiektu klasy „DashboardForm”.
* Klasa „DashboardForm” stanowi główne okno programu, które zawiera implementację większości opcji projektu. Metody, takie jak „updateAddGradesPanel”, „updateGradesTable”, „updateFinalGradesTable” oraz „updateStatsPanel” odpowiadają za odświeżenie zawartości poszczególnych komponentów wewnątrz tego okna.

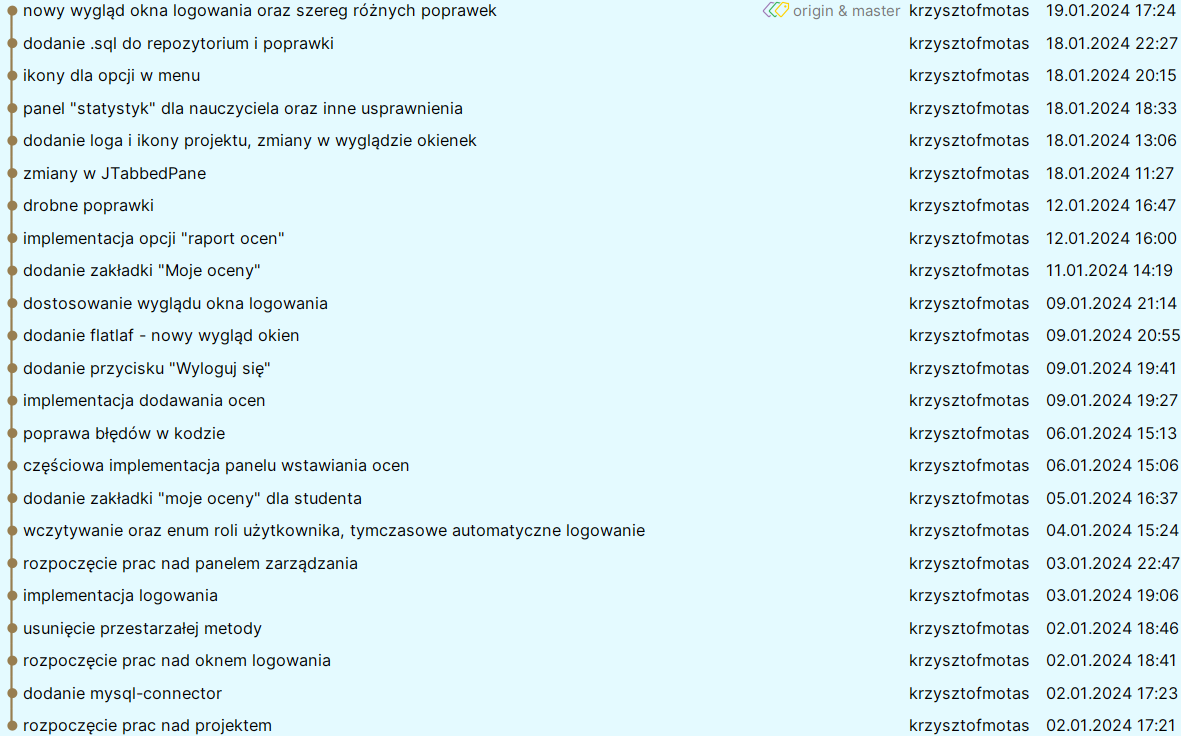
1. **Harmonogram realizacji projektu**

Projekt został zrealizowany w ciągu 13 dni. Poniżej przedstawiono diagram Gantta, uwzględniający czas poświęcony na poszczególne etapy, takie jak analiza założeń i wymagań, połączenie bazy danych z programem, tworzenie struktury bazy danych i relacji, projektowanie interfejsu użytkownika, implementacja logiki działania programu, przeprowadzenie testów aplikacji, poprawa znalezionych błędów oraz tworzenie dokumentacji. Jak widać na diagramie, najwięcej czasu poświęcono na implementację logiki działania programu.



Rysunek 3. Diagram Gantta

Projekt został zrealizowany, korzystając z systemu kontroli wersji Git. Wszystkie pliki źródłowe projektu zostały udostępnione pod poniżej podanym adresem i będą dostępne do 4 lutego 2025 roku. Link: <https://github.com/krzysztofmotas/ElektronicznySystemOceniania>. Rysunek nr 6 przedstawia zrzut ekranu historii commitów.



Rysunek 4. Historia commitów

1. **Prezentacja warstwy użytkowej projektu**

Na rysunku nr 5 przedstawiono okno informujące użytkownika o błędzie połączenia z bazą danych. Jeżeli baza danych nie została poprawnie zdefiniowana lub wystąpił inny błąd, użytkownikowi zostanie pokazane to okno.

**Obraz zawierający tekst, zrzut ekranu, Czcionka, numer

Opis wygenerowany automatycznie**

Rysunek 5. Okno błędu połączenia z bazą danych

W przypadku poprawnego połączenia z bazą danych, użytkownikowi zostanie pokazane okno logowania (rysunek nr 6), w którym użytkownik może podać swój adres e-mail oraz hasło, a następnie zalogować się do systemu. Zalogowanie jest wymagane do korzystania z elektronicznego systemu oceniania.

Obraz zawierający tekst, zrzut ekranu, Czcionka, numer

Opis wygenerowany automatycznie

Rysunek 6. Okno logowania

W przypadku, gdy użytkownik omyłkowo poda nieprawidłowy adres e-mail lub hasło, zostanie pokazane okno z odpowiednią informacją (rysunek 7 oraz rysunek 8).

Obraz zawierający tekst, zrzut ekranu, Czcionka, numer

Opis wygenerowany automatycznie

Rysunek 7. Błąd logowania

Obraz zawierający tekst, zrzut ekranu, Czcionka, numer

Opis wygenerowany automatycznie

Rysunek 8. Błąd logowania

Po zalogowaniu do systemu, użytkownikowi zostanie otwarte okno panelu zarządzania. Zakres dostępnych opcji w panelu zależy od przypisanej roli użytkownika. Rysunek nr 9 przedstawia układ panelu zarządzania dla zalogowanego nauczyciela. W prawym górnym rogu okna widoczne są informacje o zalogowanym użytkowniku oraz przycisk służący do wylogowania z panelu zarządzania. W centrum okna znajduje się menu zawierające opcje takie jak "Wstawianie ocen", "Raport ocen" oraz "Statystyki".

Na zrzucie ekranu jest zaznaczona opcja "Wstawianie ocen", umożliwiająca nauczycielowi przypisanie nowych ocen swoim uczniom. Po lewej stronie panelu znajdują się wymagane opcje wyboru przedmiotu i rodzaju oceny. Po prawej stronie panelu znajduje się tabela zawierająca listę wszystkich studentów. Dla każdego studenta możliwe jest przypisanie oceny. Po uzupełnieniu ocen dla wybranych studentów, nauczyciel potwierdza swoją decyzję za pomocą przycisku "Wstaw oceny" znajdującego się w prawym dolnym rogu okna.

Obraz zawierający tekst, zrzut ekranu, wyświetlacz, oprogramowanie

Opis wygenerowany automatycznie

Rysunek 9. Wygląd panelu zarządzania dla nauczyciela

Na rysunku nr 10 przedstawiono proces uzupełniania ocen

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, wyświetlacz

Opis wygenerowany automatycznie

Rysunek 10. Proces uzupełniania ocen

Po zatwierdzeniu decyzji przyciskiem „Wstaw oceny”, użytkownikowi zostanie pokazane okno przedstawione na rysunku nr 11.

Obraz zawierający tekst, zrzut ekranu, Czcionka, numer

Opis wygenerowany automatycznie

Rysunek 11. Informacja o dodaniu ocen

Wybierając opcję „Raport ocen” użytkownikowi zostanie pokazana zawartość panelu tej opcji (rysunek nr 12). W lewej stronie panelu znajdują się pola tekstowe, umożliwiające podanie imienia i nazwiska studenta, dla którego raport ocen ma zostać wygenerowany. Po prawej stronie panelu znajdzie się tabela ocen wyszukanego studenta.

Obraz zawierający tekst, zrzut ekranu, wyświetlacz, oprogramowanie

Opis wygenerowany automatycznie

Rysunek 12. Raport ocen

Po uzupełnieniu pól tekstowych oraz potwierdzenia wygenerowania raportu za pomocą przycisku „Wyszukaj oceny”, zostanie wygenerowany raport w postaci tabeli (rysunek nr 13), znajdującej się po prawej stronie panelu. W tabeli znajdują się takie kolumny jak przedmiot, oceny oraz średnia ocen.

Obraz zawierający tekst, zrzut ekranu, wyświetlacz, oprogramowanie

Opis wygenerowany automatycznie

Rysunek 13. Wygenerowany raport ocen

Wybierając opcję „Statystyki” użytkownikowi zostanie pokazany panel statystyk (rysunek nr 14), zawierający tabelę ze statystykami. Tabela ta obejmuje informacje dotyczące wszystkich przedmiotów, takie jak najniższa i najwyższa średnia ocen, średnia wszystkich ocen oraz mediana.

Obraz zawierający tekst, zrzut ekranu, wyświetlacz, oprogramowanie

Opis wygenerowany automatycznie

Rysunek 14. Panel statystyk

Na rysunku nr 15 przedstawiono wygląd panelu zarządzania dla zalogowanego studenta. Dla studenta w menu dostępne są takie opcje jak „Moje oceny” oraz „Oceny końcowe”. W panelu „Moje oceny” widoczna jest tabela zawierająca informacje o ocenach zalogowanego studenta.

Obraz zawierający tekst, zrzut ekranu, wyświetlacz, oprogramowanie

Opis wygenerowany automatycznie

Rysunek 15. Panel zarządzania w przypadku studenta

W przypadku wybrania opcji „Oceny końcowe” użytkownik ma możliwość podglądu średniej ocen oraz ocen końcowych z wszystkich przedmiotów.

Obraz zawierający tekst, zrzut ekranu, wyświetlacz, oprogramowanie

Opis wygenerowany automatycznie

Rysunek 16. Panel ocen końcowych

1. **Podsumowanie**

Realizacja projektu przebiegła pomyślnie, a stworzony program działa zgodnie z założeniami. Udało się zrealizować wszystkie założenia, tworząc przy tym intuicyjną aplikację. W przyszłości projekt można rozbudować o dodatkowe funkcje, takie jak podział studentów według kierunku, generowanie wykresów ocen oraz rozbudowany system oceniania dla nauczycieli, uwzględniający więcej opcji. Te rozszerzenia mogą zwiększyć funkcjonalność i użyteczność programu, dostarczając użytkownikom jeszcze więcej opcji i narzędzi do zarządzania danymi.

Projekt wydaje się solidną podstawą, a zaproponowane rozszerzenia mogą zwiększyć użyteczność aplikacji i dostosować ją do rosnących potrzeb użytkowników.

1. **Literatura**
2. <https://regexr.com/3e48o> (data dostępu: 3.01.2024)
3. <https://stackoverflow.com/questions/19605150/regex-for-password-must-contain-at-least-eight-characters-at-least-one-number-a> (data dostępu: 4.01.2024)
4. <https://stackoverflow.com/questions/17627431/auto-resizing-the-jtable-column-widths> (data dostępu: 12.01.2024)
5. <https://stackoverflow.com/questions/144892/how-to-center-a-window-in-java> (data dostępu: 18.01.2024)
6. <https://www.svgrepo.com/> (data dostępu: 18.01.2024)
7. <https://github.com/weisJ/jsvg> (data dostępu: 18.01.2024)
8. <https://github.com/JFormDesigner/FlatLaf> (data dostępu: 9.01.2024)